1 Pseudo Three-dimensional Surface Code

In this section, we describe the pseudo three-dimensional Surface Code on the looped pipeline architecture introduced in Section **??**. First, we describe how computation is performed on multiple 2D Surface Codes in a processor. Then, we extend this concept into a pseudo three-dimensional structure with a periodic cycle in the direction of the third dimension.

1.1 Quantum Processor

In fault-tolerant quantum computation, the Surface Code, introduced in Section **??**, is the most promising error correction code for the calculations required in many quantum algorithms. On the other hand, quantum low-density parity-check codes (qLDPC) are often considered more suitable for quantum memory due to their high encoding rate. However, while a single Surface Code can encode only one logical qubit, it offers many advantages, such as a simple approach for universal logical operations using lattice surgery combined with magic state distillation.

When designing the processor for computation, we simplify a single Surface Code into a "patch," which features dashed and solid lines. Simply put, a patch represents a logical qubit. In Fig. 1(a), three patches are allocated on the processor, and the corresponding Surface Codes are shown in Fig. 1(b), which are numbered. The rest of the qubits in Fig. 1(b) are unused data qubits for lattice surgery, as introduced in Section **??**.

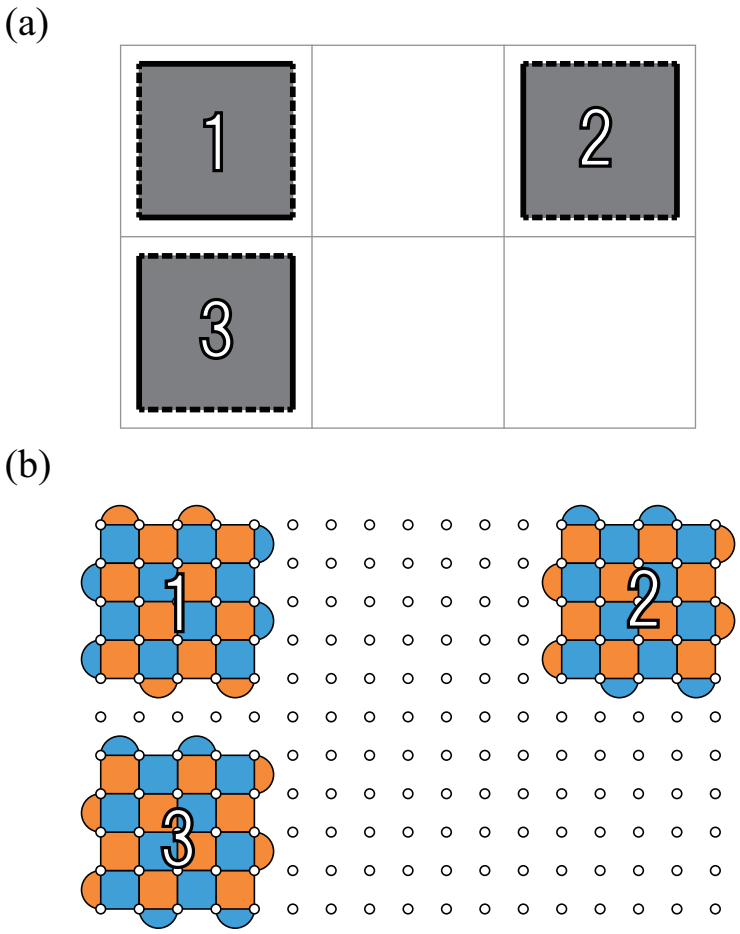

Fig. 1

Using lattice surgery, we can perform logical operations between two patches, three patches, or more. Additionally, we can perform commutative surgery operations in parallel when there exists a route from the control qubit to the target qubit by using unused data qubits in the processor. In this scheme, the efficiency of computation depends on how many parallel operations we can execute, thus requiring careful decision-making regarding the routing of operations. For instance, a certain logical 2-qubit operation between patch 2 and patch 3 is shown in Fig. 2. In this case, we cannot perform a logical operation between patch 1 and patch 2 in parallel.

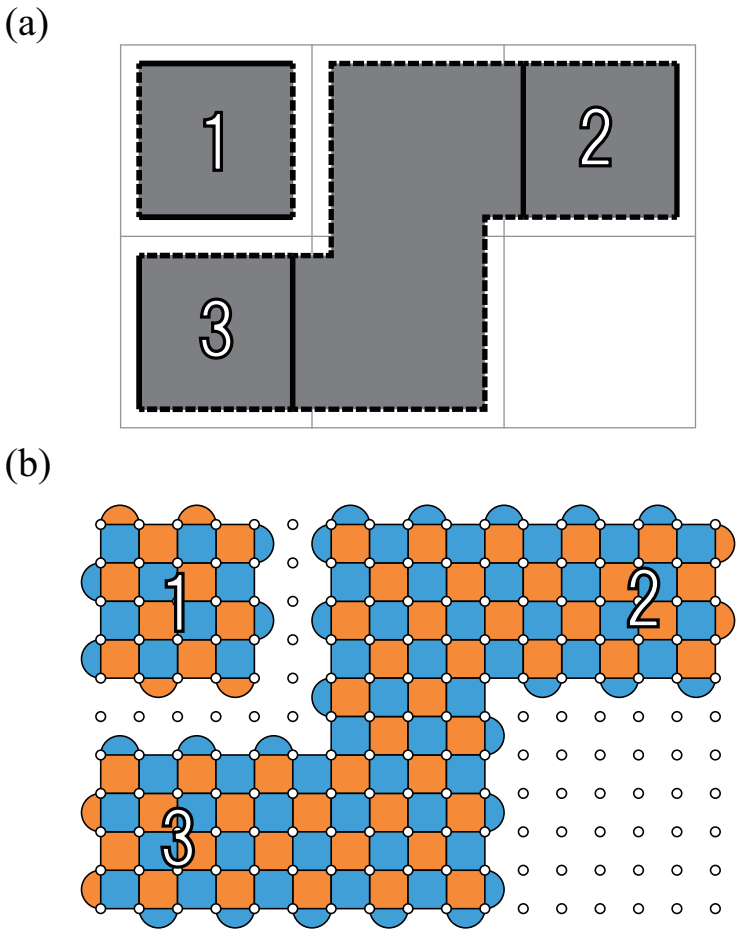

Fig. 2

In Fig. 3, some logical operations performed in parallel are illustrated with blue routes, while the orange route is prohibited since it intersects with an existing blue route at their intersection.

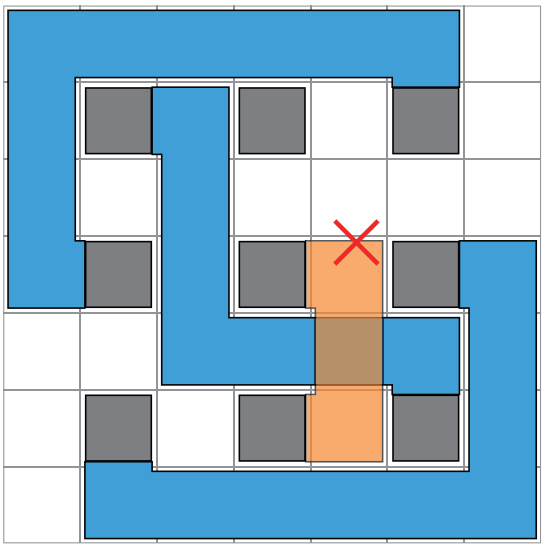

Fig. 3